

Ref #	Hits	Search Query	DBs	Default Operator	Plurals	Time Stamp
L1	1	"5930811".pn.	US-PGPUB; USPAT	OR	OFF	2005/10/14 13:59
L2	46216	dictionary	US-PGPUB; USPAT	OR	OFF	2005/10/14 13:59
L3	69126	dictionary	US-PGPUB; USPAT; EPO; JPO; DERWENT	OR	OFF	2005/10/14 13:59
L4	2659	dictionary & keyword	US-PGPUB; USPAT; EPO; JPO; DERWENT	OR	OFF	2005/10/14 13:59
L5	27980	dictionary & input	US-PGPUB; USPAT; EPO; JPO; DERWENT	OR	OFF	2005/10/14 13:59
L6	1718	(dictionary & input) & "word processor"	US-PGPUB; USPAT; EPO; JPO; DERWENT	OR	OFF	2005/10/14 13:59
L7	147	((dictionary & input) & "word processor") & synonyms	US-PGPUB; USPAT; EPO; JPO; DERWENT	OR	OFF	2005/10/14 13:59
L8	4634	(dictionary & input) & (character near2 input)	US-PGPUB; USPAT; EPO; JPO; DERWENT	OR	OFF	2005/10/14 13:59
L9	287	((dictionary & input) & (character near2 input)) & keyword	US-PGPUB; USPAT; EPO; JPO; DERWENT	OR	OFF	2005/10/14 13:59
L10	84	((((dictionary & input) & (character near2 input)) & keyword) & menu	US-PGPUB; USPAT; EPO; JPO; DERWENT	OR	OFF	2005/10/14 13:59
L11	81	(((((dictionary & input) & (character near2 input)) & keyword) & menu) & display	US-PGPUB; USPAT; EPO; JPO; DERWENT	OR	OFF	2005/10/14 13:59
L12	9	((((((dictionary & input) & (character near2 input)) & keyword) & menu) & display) & "search request"	US-PGPUB; USPAT; EPO; JPO; DERWENT	OR	OFF	2005/10/14 13:59
L13	2140	dictionary & "electronic dictionary"	US-PGPUB; USPAT; EPO; JPO; DERWENT	OR	OFF	2005/10/14 13:59

L14	1093	(dictionary & "electronic dictionary") & input	US-PGPUB; USPAT; EPO; JPO; DERWENT	OR	OFF	2005/10/14 13:59
L15	103	((dictionary & "electronic dictionary") & input) & keyword	US-PGPUB; USPAT; EPO; JPO; DERWENT	OR	OFF	2005/10/14 13:59
L16	34	((((dictionary & "electronic dictionary") & input) & keyword) & dictionaries	US-PGPUB; USPAT; EPO; JPO; DERWENT	OR	OFF	2005/10/14 13:59
L17	226	715/532.ccls.	US-PGPUB; USPAT	OR	OFF	2005/10/14 13:59
L18	5	"5930811".URPN.	USPAT	OR	OFF	2005/10/14 13:59
L19	14	("4454576" "4800510" "5379373" "5517621" "5649216" "5649220" "5845303" "5900002" "5930811" "6009431" "6016146" "6144974" "6182096" "6301586").PN.	USPAT	OR	OFF	2005/10/14 13:59
L20	5	("5424945" "5438512" "5517621" "5666289" "5669007").PN.	USPAT	OR	OFF	2005/10/14 13:59
L21	1	"5983237".pn.	US-PGPUB; USPAT	OR	OFF	2005/10/14 13:59
L22	46216	(US "20020007382" A1 US "6381598" B1 US "5983237" A) & dictionary	US-PGPUB; USPAT	OR	OFF	2005/10/14 13:59
L23	1	"6381598".pn. & plurality	US-PGPUB; USPAT	OR	OFF	2005/10/14 13:59
L24	1	"6381598".pn. & (plurality Near US-6381598-B1.DID. dictionar\$)	US-PGPUB; USPAT	OR	OFF	2005/10/14 13:59
L25	0	"6381598".pn. & (plurality Near4 dictionar\$)	US-PGPUB; USPAT	OR	OFF	2005/10/14 13:59
L26	1	"6381598".pn. & dictionaries	US-PGPUB; USPAT	OR	OFF	2005/10/14 13:59
L27	2	"6665838".pn. "6567800".pn.	US-PGPUB; USPAT	OR	OFF	2005/10/14 13:59
L28	30	gaelic	US-PGPUB; USPAT	OR	OFF	2005/10/14 13:59
L29	4	gaelic & dictionar\$	US-PGPUB; USPAT	OR	OFF	2005/10/14 13:59
L30	245959	cancel near "3" button	US-PGPUB; USPAT	OR	OFF	2005/10/14 13:59
L31	5456	"cancel button"	US-PGPUB; USPAT	OR	OFF	2005/10/14 13:59

L32	265	"cancel button" & "search engine"	US-PGPUB; USPAT	OR	OFF	2005/10/14 13:59
L33	58	("cancel button" & "search engine") & dictionar\$	US-PGPUB; USPAT	OR	OFF	2005/10/14 13:59
L34	46	((("cancel button" & "search engine") & dictionar\$) & menu	US-PGPUB; USPAT	OR	OFF	2005/10/14 13:59
L35	2	((("cancel button" & "search engine") & dictionar\$) & menu) & "menu screen"	US-PGPUB; USPAT	OR	OFF	2005/10/14 13:59
L36	58	sjis	US-PGPUB; USPAT	OR	OFF	2005/10/14 13:59
L37	52	sjis & format	US-PGPUB; USPAT	OR	OFF	2005/10/14 13:59
L38	15	(sjis & format) & keyword	US-PGPUB; USPAT	OR	OFF	2005/10/14 13:59
L39	147	"invalid format"	US-PGPUB; USPAT	OR	OFF	2005/10/14 13:59
L40	1	("invalid format" & menu) & keyword	US-PGPUB; USPAT	OR	OFF	2005/10/14 13:59
L41	59	"invalid format" & menu	US-PGPUB; USPAT	OR	OFF	2005/10/14 13:59
L42	8	"invalid format" & keyword	US-PGPUB; USPAT	OR	OFF	2005/10/14 13:59
L43	1312	"cell format"	US-PGPUB; USPAT	OR	OFF	2005/10/14 13:59
L44	272	"cell format" & (invalid bad)	US-PGPUB; USPAT	OR	OFF	2005/10/14 13:59
L45	0	("cell format" & (invalid bad)) & ((invalid bad) ADJ format)	US-PGPUB; USPAT	OR	OFF	2005/10/14 13:59
L46	0	("cell format" & (invalid bad)) & ((invalid bad) ADJ3 format)	US-PGPUB; USPAT	OR	OFF	2005/10/14 13:59
L47	134427	format & error	US-PGPUB; USPAT	OR	OFF	2005/10/14 13:59
L48	115	"invalid format" & error	US-PGPUB; USPAT	OR	OFF	2005/10/14 13:59
L49	1	("invalid format" & error) & keyword	US-PGPUB; USPAT	OR	OFF	2005/10/14 13:59
L50	39	("invalid format" & error) & "error message"	US-PGPUB; USPAT	OR	OFF	2005/10/14 13:59
L51	23799	error same format	US-PGPUB; USPAT	OR	OFF	2005/10/14 13:59
L52	23	((("invalid format" & error) & "error message") & (error same format)	US-PGPUB; USPAT	OR	OFF	2005/10/14 13:59
L53	20	"4012725".URPN.	USPAT	OR	OFF	2005/10/14 13:59
L54	9750	"search engine"	US-PGPUB; USPAT	OR	OFF	2005/10/14 13:59

L55	9	"search engine" & (invalid bad) ADJ format	US-PGPUB; USPAT	OR	OFF	2005/10/14 13:59
L56	0	"09665635".an.	US-PGPUB; USPAT	OR	OFF	2005/10/14 13:59
L57	0	"09665635".apn.	US-PGPUB; USPAT	OR	OFF	2005/10/14 13:59
L58	1617	kana	US-PGPUB; USPAT	OR	OFF	2005/10/14 13:59
L59	523	L58 & dictionary\$	US-PGPUB; USPAT	OR	OFF	2005/10/14 13:59
L60	77	L59 & keyword\$	US-PGPUB; USPAT	OR	OFF	2005/10/14 13:59
S1	1	"5930811".pn.	US-PGPUB; USPAT	OR	OFF	2004/03/03 14:52
S2	36871	dictionary	US-PGPUB; USPAT	OR	OFF	2004/03/03 14:53
S3	57600	dictionary	US-PGPUB; USPAT; EPO; JPO; DERWENT	OR	OFF	2004/03/03 14:59
S4	1929	dictionary & keyword	US-PGPUB; USPAT; EPO; JPO; DERWENT	OR	OFF	2004/03/03 15:00
S5	22909	dictionary & input	US-PGPUB; USPAT; EPO; JPO; DERWENT	OR	OFF	2004/03/03 15:00
S6	1461	(dictionary & input) & "word processor"	US-PGPUB; USPAT; EPO; JPO; DERWENT	OR	OFF	2004/03/03 15:03
S7	107	((dictionary & input) & "word processor") & synonyms	US-PGPUB; USPAT; EPO; JPO; DERWENT	OR	OFF	2004/03/03 15:03
S8	4190	(dictionary & input) & (character near2 input)	US-PGPUB; USPAT; EPO; JPO; DERWENT	OR	OFF	2004/03/03 15:04
S9	234	((dictionary & input) & (character near2 input)) & keyword	US-PGPUB; USPAT; EPO; JPO; DERWENT	OR	OFF	2004/03/03 15:05
S10	75	((((dictionary & input) & (character near2 input)) & keyword) & menu	US-PGPUB; USPAT; EPO; JPO; DERWENT	OR	OFF	2004/03/03 15:05

S11	72	(((((dictionary & input) & (character near2 input)) & keyword) & menu) & display	US-PGPUB; USPAT; EPO; JPO; DERWENT	OR	OFF	2004/03/03 15:06
S12	8	(((((dictionary & input) & (character near2 input)) & keyword) & menu) & display) & "search request"	US-PGPUB; USPAT; EPO; JPO; DERWENT	OR	OFF	2004/03/03 15:37
S13	1429	dictionary & "electronic dictionary"	US-PGPUB; USPAT; EPO; JPO; DERWENT	OR	OFF	2004/03/03 15:10
S14	775	(dictionary & "electronic dictionary") & input	US-PGPUB; USPAT; EPO; JPO; DERWENT	OR	OFF	2004/03/03 15:10
S15	66	((dictionary & "electronic dictionary") & input) & keyword	US-PGPUB; USPAT; EPO; JPO; DERWENT	OR	OFF	2004/03/03 15:10
S16	20	(((((dictionary & "electronic dictionary") & input) & keyword) & dictionaries	US-PGPUB; USPAT; EPO; JPO; DERWENT	OR	OFF	2004/03/03 15:10
S17	172	715/532.ccls.	US-PGPUB; USPAT	OR	OFF	2004/03/03 15:37
S18	3	"5930811".URPN.	USPAT	OR	OFF	2004/03/23 08:47
S19	14	("4454576" "4800510" "5379373" "5517621" "5649216" "5649220" "5845303" "5900002" "5930811" "6009431" "6016146" "6144974" "6182096" "6301586").PN.	USPAT	OR	OFF	2004/03/23 08:47
S20	5	("5424945" "5438512" "5517621" "5666289" "5669007").PN.	USPAT	OR	OFF	2004/03/23 08:48
S21	1	"5983237".pn.	US-PGPUB; USPAT	OR	OFF	2004/03/23 10:04
S22	37187	(US "20020007382" A1 US "6381598" B1 US "5983237" A) & dictionary	US-PGPUB; USPAT	OR	OFF	2004/03/23 10:10
S23	1	"6381598".pn. & plurality	US-PGPUB; USPAT	OR	OFF	2004/03/23 10:12
S24	1	"6381598".pn. & (plurality Near US-6381598-B1.DID. dictionar\$)	US-PGPUB; USPAT	OR	OFF	2004/03/23 10:13
S25	0	"6381598".pn. & (plurality Near4 dictionar\$)	US-PGPUB; USPAT	OR	OFF	2004/03/23 10:13

S26	1	"6381598".pn. & dictionaries	US-PGPUB; USPAT	OR	OFF	2004/03/23 10:13
S27	2	"6665838".pn. "6567800".pn.	US-PGPUB; USPAT	OR	OFF	2004/03/23 10:27
S28	23	gaelic	US-PGPUB; USPAT	OR	OFF	2004/03/23 10:27
S29	1	gaelic & dictionar\$	US-PGPUB; USPAT	OR	OFF	2004/03/23 10:51
S30	194286	cancel near "3" button	US-PGPUB; USPAT	OR	OFF	2004/03/23 10:52
S31	3738	"cancel button"	US-PGPUB; USPAT	OR	OFF	2004/03/23 10:52
S32	171	"cancel button" & "search engine"	US-PGPUB; USPAT	OR	OFF	2004/03/23 10:52
S33	40	("cancel button" & "search engine") & dictionar\$	US-PGPUB; USPAT	OR	OFF	2004/03/23 10:53
S34	31	((("cancel button" & "search engine") & dictionar\$) & menu	US-PGPUB; USPAT	OR	OFF	2004/03/23 10:53
S35	2	((("cancel button" & "search engine") & dictionar\$) & menu) & "menu screen"	US-PGPUB; USPAT	OR	OFF	2004/03/23 10:53
S36	45	sjis	US-PGPUB; USPAT	OR	OFF	2004/03/23 11:21
S37	41	sjis & format	US-PGPUB; USPAT	OR	OFF	2004/03/23 11:22
S38	10	(sjis & format) & keyword	US-PGPUB; USPAT	OR	OFF	2004/03/23 11:28
S39	124	"invalid format"	US-PGPUB; USPAT	OR	OFF	2004/03/23 11:30
S40	0	("invalid format" & menu) & keyword	US-PGPUB; USPAT	OR	OFF	2004/03/23 11:28
S41	51	"invalid format" & menu	US-PGPUB; USPAT	OR	OFF	2004/03/23 11:29
S42	4	"invalid format" & keyword	US-PGPUB; USPAT	OR	OFF	2004/03/23 11:32
S43	1067	"cell format"	US-PGPUB; USPAT	OR	OFF	2004/03/23 11:32
S44	211	"cell format" & (invalid bad)	US-PGPUB; USPAT	OR	OFF	2004/03/23 11:32
S45	0	("cell format" & (invalid bad)) & ((invalid bad) ADJ format)	US-PGPUB; USPAT	OR	OFF	2004/03/23 11:34
S46	0	("cell format" & (invalid bad)) & ((invalid bad) ADJ3 format)	US-PGPUB; USPAT	OR	OFF	2004/03/23 11:34
S47	100358	format & error	US-PGPUB; USPAT	OR	OFF	2004/03/23 11:35

S48	100	"invalid format" & error	US-PGPUB; USPAT	OR	OFF	2004/03/23 11:36
S49	0	("invalid format" & error) & keyword	US-PGPUB; USPAT	OR	OFF	2004/03/23 11:36
S50	33	("invalid format" & error) & "error message"	US-PGPUB; USPAT	OR	OFF	2004/03/23 11:38
S51	18439	error same format	US-PGPUB; USPAT	OR	OFF	2004/03/23 11:37
S52	18	((("invalid format" & error) & "error message") & (error same format)	US-PGPUB; USPAT	OR	OFF	2004/03/23 11:38
S53	20	"4012725".URPN.	USPAT	OR	OFF	2004/03/23 11:45
S54	5969	"search engine"	US-PGPUB; USPAT	OR	OFF	2004/03/23 12:07
S55	6	"search engine" & (invalid bad) ADJ format	US-PGPUB; USPAT	OR	OFF	2004/03/23 12:08
S56	0	"09665635".an.	US-PGPUB; USPAT	OR	OFF	2004/03/24 10:18
S57	0	"09665635".apn.	US-PGPUB; USPAT	OR	OFF	2004/03/24 10:18
S58	1465	kana	US-PGPUB; USPAT	OR	OFF	2004/12/21 17:49
S59	479	S58 & dictionary\$	US-PGPUB; USPAT	OR	OFF	2004/12/21 17:49
S60	65	S59 & keyword\$	US-PGPUB; USPAT	OR	OFF	2004/12/21 17:49



USPTO

[Subscribe](#) (Full Service) [Register](#) (Limited Service, Free) [Login](#)

 Search: ☒ The ACM Digital Library ☐ The Guide



THE ACM DIGITAL LIBRARY


[Feedback](#) [Report a problem](#) [Satisfaction survey](#)

Published before July 2000

Terms used **search string input keyword dictionary**Found **5,601** of 109,665

Sort results by

relevance

Display results

expanded form

☒ [Save results to a Binder](#)
☒ [Search Tips](#)
☐ Open results in a new window
Try an [Advanced Search](#)Try this search in [The ACM Guide](#)

Results 1 - 20 of 200

Result page: [1](#) [2](#) [3](#) [4](#) [5](#) [6](#) [7](#) [8](#) [9](#) [10](#) [next](#)

Best 200 shown

Relevance scale ☐ ☐ ☐ ☐ ☐

1 [Technique for automatically correcting words in text](#)

Karen Kukich

December 1992 **ACM Computing Surveys (CSUR)**, Volume 24 Issue 4

Full text available: pdf(6.23 MB)

 Additional Information: [full citation](#), [abstract](#), [references](#), [citations](#), [index terms](#), [review](#)

Research aimed at correcting words in text has focused on three progressively more difficult problems: (1) nonword error detection; (2) isolated-word error correction; and (3) context-dependent word correction. In response to the first problem, efficient pattern-matching and n-gram analysis techniques have been developed for detecting strings that do not appear in a given word list. In response to the second problem, a variety of general and application-specific spelling cor ...

Keywords: n-gram analysis, Optical Character Recognition (OCR), context-dependent spelling correction, grammar checking, natural-language-processing models, neural net classifiers, spell checking, spelling error detection, spelling error patterns, statistical-language models, word recognition and correction

2 [Simple word strings as compound keywords: an indexing and ranking method for Japanese texts](#)

Yasushi Ogawa, Ayako Bessho, Masako Hirose

July 1993 **Proceedings of the 16th annual international ACM SIGIR conference on Research and development in information retrieval**

Full text available: pdf(1.06 MB)


 Additional Information: [full citation](#), [abstract](#), [references](#), [citations](#), [index terms](#), [review](#)

This paper describes a new indexing method for Japanese text databases using the simple keyword string, in which a compound word is treated as a string of simple words, which are the smallest units in Japanese grammar which still maintain their meanings. This method allows retrieved texts to be ranked, according to the similarity of their meaning to the query, without using a control vocabulary or thesaurus. This paper also introduces the keyword feature, w ...

3 [Modeling for text compression](#)

Timothy Bell, Ian H. Witten, John G. Cleary

December 1989 **ACM Computing Surveys (CSUR)**, Volume 21 Issue 4

Full text available:  [pdf\(3.54 MB\)](#)


Additional Information: [full citation](#), [abstract](#), [references](#), [citations](#), [index terms](#), [review](#)

The best schemes for text compression use large models to help them predict which characters will come next. The actual next characters are coded with respect to the prediction, resulting in compression of information. Models are best formed adaptively, based on the text seen so far. This paper surveys successful strategies for adaptive modeling that are suitable for use in practical text compression systems. The strategies fall into three main classes: finite-context modeling, i ...

4 [An efficient text input method for pen-based computers](#)

Toshiyuki Masui

January 1998 **Proceedings of the SIGCHI conference on Human factors in computing systems**

Full text available:  [pdf\(1.06 MB\)](#)


Additional Information: [full citation](#), [references](#), [citations](#), [index terms](#)

Keywords: POBox, hand-held devices, input devices, international interfaces, pen-based input, predictive interface

5 [Fast detection of communication patterns in distributed executions](#)

Thomas Kunz, Michiel F. H. Seuren

November 1997 **Proceedings of the 1997 conference of the Centre for Advanced Studies on Collaborative research**

Full text available:  [pdf\(4.21 MB\)](#)

Additional Information: [full citation](#), [abstract](#), [references](#), [index terms](#)

Understanding distributed applications is a tedious and difficult task. Visualizations based on process-time diagrams are often used to obtain a better understanding of the execution of the application. The visualization tool we use is Poet, an event tracer developed at the University of Waterloo. However, these diagrams are often very complex and do not provide the user with the desired overview of the application. In our experience, such tools display repeated occurrences of non-trivial commun ...

6 [The Least Common Bigram: A dictionary arrangement technique for computerized natural-language text searching](#)

Elaine M. Onderisin

January 1971 **Proceedings of the 1971 26th annual conference**

Full text available:  [pdf\(761.47 KB\)](#)

Additional Information: [full citation](#), [abstract](#), [references](#), [index terms](#)


The LCB (Least Common Bigram) method is a dictionary arrangement technique for computerized natural-language searching, based upon certain statistical characteristics of a data base. Once the distribution of elements (such as bigrams) in the data base is known, the LCB method allows each dictionary entry to be sorted on the element least likely to be found in the data base. Thus the LCB method minimizes the number of dictionary entries examined during text searching. Statistical analysis of ...

Keywords: Bigram, Dictionary lookup, Digram, Free text searching, Information retrieval, Literature searching, Natural language text searching, SDI, Screen search

7 [From text to hypertext by indexing](#)

Airi Salminen, Jean Tague-Sutcliffe, Charles McClellan

January 1995 **ACM Transactions on Information Systems (TOIS)**, Volume 13 Issue 1

Full text available:  [pdf\(1.98 MB\)](#)

Additional Information: [full citation](#), [abstract](#), [references](#), [citations](#), [index](#)

[terms](#), [review](#)


A model is presented for converting a collection of documents to hypertext by means of indexing. The documents are assumed to be semistructured, i.e., their text is a hierarchy of parts, and some of the parts consist of natural language. The model is intended as a framework for specifying hypertextual reading capabilities for specific application areas and for developing new automated tools for the conversion of semistructured text to hypertext. In the model, two well-known paradigms— ...

Keywords: constrained grammars, grammars, hypertext, properties, structured text, test types, text entities, transient hypergraphs

8 [A general interactive guidance for information retrieval and processing systems](#)

R. Erbe, G. Walch

September 1976 **Proceedings of the eighth international conference on APL**


Full text available:  [pdf\(959.24 KB\)](#) Additional Information: [full citation](#), [abstract](#), [references](#), [citations](#), [index terms](#)

A general application guidance system -GAGS- was developed to support the problem solving enduser in applying DP-facilities without being a DP-professional. It has the following four main features: 1. Dialogue supported methods dictionary: guides the user from a given application problem to the adequate solution method or program. 2. Dialogue supported data dictionary: gives syntactic and semantic information about the data available in data administration systems such ...

9 [A very fast substring search algorithm](#)

Daniel M. Sunday

August 1990 **Communications of the ACM**, Volume 33 Issue 8


Full text available:  [pdf\(1.03 MB\)](#) Additional Information: [full citation](#), [abstract](#), [references](#), [citations](#), [index terms](#), [review](#)

This article describes a substring search algorithm that is faster than the Boyer-Moore algorithm. This algorithm does not depend on scanning the pattern string in any particular order. Three variations of the algorithm are given that use three different pattern scan orders. These include: (1) a "Quick Search" algorithm; (2) a "Maximal Shift" and (3) an "Optimal Mismatch" algorithm.

10 [Tools for visualizing text compression algorithms](#)

Sami Khuri, Hsiu-Chin Hsu

March 2000 **Proceedings of the 2000 ACM symposium on Applied computing - Volume 1**



Full text available:  [pdf\(561.39 KB\)](#) Additional Information: [full citation](#), [references](#), [index terms](#)

Keywords: adaptive Huffman coding, algorithms, dictionary encoding, text compression, visualization

11 [The FINITE STRING newsletter: Abstracts of current literature](#)

American Journal of Computational Linguistics Staff

April 1983 **Computational Linguistics**, Volume 9 Issue 2

Full text available:  [pdf\(2.03 MB\)](#)  Additional Information: [full citation](#)
[Publisher Site](#)

12 Natural language in document retrieval systems: CUE: a preprocessor system for restricted, natural English

David B. Loveman, John A. Moyné, Robert G. Tobey

April 1971 **Proceedings of the 1971 international ACM SIGIR conference on Information storage and retrieval**

Full text available:  [pdf\(964.79 KB\)](#) Additional Information: [full citation](#), [abstract](#), [references](#)

CUE, an input interface system which permits the computer to utilize natural but restricted English as input, is presented. In addition, an experimental model for CUE, Proto-RELADES, which can "understand" and execute English sentences about the content of the library at IBM's Boston Programming Center is described. These sentences can be query, command, or conditional sentences. The linguistic component of the system is based on a transformational grammar of English that performs a full syntact ...

Keywords: CUE, English, RELADES, computational linguistics, grammar, natural language, parsing, semantics, syntax

13 Comparative analysis of hardware versus software text search

Peter Kracsony, Gerald Kowalski, Arnold Meltzer


June 1980 **Proceedings of the 3rd annual ACM conference on Research and development in information retrieval**

Full text available:  [pdf\(515.52 KB\)](#) Additional Information: [full citation](#), [references](#), [citations](#)

14 Spelling correction in user interfaces

Ivor Durham, David A. Lamb, James B. Saxe

October 1983 **Communications of the ACM**, Volume 26 Issue 10

Full text available:  [pdf\(1.12 MB\)](#) Additional Information: [full citation](#), [abstract](#), [references](#), [citations](#), [index terms](#)

The feasibility of providing a spelling corrector as a part of interactive user interfaces is demonstrated. The issues involved in using spelling correction in a user interface are examined, and a simple correction algorithm is described. The results of an experiment in which the corrector is incorporated into a heavily used interactive program are described. More than one quarter of the errors made by users during the experiment were corrected using the simple mechanisms presented ...

Keywords: interactive programs, spelling, spelling correction, typographical errors, user interfaces

15 An approach to integration of Web information source search and Web information retrieval

Yuichi Lizuka, Mitsuaki Tsunakawa, Shin-ichiro Seo, Tetsuo Ikeda

March 2000 **Proceedings of the 2000 ACM symposium on Applied computing - Volume 1**

Full text available:  [pdf\(807.66 KB\)](#) Additional Information: [full citation](#), [references](#), [citations](#), [index terms](#)

Keywords: WWW, heterogeneous information sources, information resource management, mediator, universal relation

16 Natural language in document retrieval systems: Full text document retrieval: Hebrew legal texts (report on the first phase of the responsa retrieval project)

Y. Choueka, M. Cohen, J. Dueck, A. S. Fraenkel, M. Slae

April 1971 **Proceedings of the 1971 international ACM SIGIR conference on Information storage and retrieval**

Full text available:  pdf(1.25 MB) Additional Information: [full citation](#), [abstract](#), [references](#), [citations](#)

A full text retrieval system was designed for the responsa literature, which is a large corpus of Hebrew legal cases. The unique problems of the data base --- mixture of Hebrew, Aramaic and vernaculars, lack of vowels and punctuation, extreme language inflection problems, homographs, existence of thousands of grammatical variants of any given keyword --- dictated development of new methods. Among them we list "grammatical synthesis", which synthesizes all grammatical variants of a given keyword; ...

Keywords: Hebrew computational linguistics, case law retrieval, feedback, full text retrieval, grammatical synthesis, legal cases, metrical operators, responsa

17 Natural language question-answering systems: 1969

Robert F. Simmons

January 1970 **Communications of the ACM**, Volume 13 Issue 1


Full text available:  pdf(2.15 MB) Additional Information: [full citation](#), [abstract](#), [references](#), [citations](#)

Recent experiments in programming natural language question-answering systems are reviewed to summarize the methods that have been developed for syntactic, semantic, and logical analysis of English strings. It is concluded that at least minimally effective techniques have been devised for answering questions from natural language subsets in small scale experimental systems and that a useful paradigm has evolved to guide research efforts in the field. Current approaches to semantic analysis ...

Keywords: artificial intelligence, fact retrieval, language processing, natural language, question-answering system, semantics

18 Conference abstracts

January 1977 **Proceedings of the 5th annual ACM computer science conference**


Full text available:  pdf(3.14 MB) Additional Information: [full citation](#), [abstract](#), [index terms](#)

One problem in computer program testing arises when errors are found and corrected after a portion of the tests have run properly. How can it be shown that a fix to one area of the code does not adversely affect the execution of another area? What is needed is a quantitative method for assuring that new program modifications do not introduce new errors into the code. This model considers the retest philosophy that every program instruction that could possibly be reached and tested from the ...

19 Optimization of a hierarchical file organization for spelling correction

Tetsuro Ito, Clement T. Yu

June 1985 **Proceedings of the 8th annual international ACM SIGIR conference on Research and development in information retrieval**

Full text available:  pdf(511.78 KB) Additional Information: [full citation](#), [abstract](#), [references](#), [citations](#)

A spelling program using a hierarchically organized file seems to be promising, since it can correct more than common typing mistakes. However, its speed of detecting spelling errors in the inputs is rather slow. Here some techniques of modifying the program to improve the speed are presented.

20 On Chinese text retrieval

Jian-Yun Nie, Martin Brisebois, Xiaobo Ren

August 1996 **Proceedings of the 19th annual international ACM SIGIR conference on Research and development in information retrieval**

Full text available:  pdf(925.66 KB) Additional Information: [full citation](#), [references](#), [citations](#), [index terms](#)



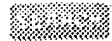
Results 1 - 20 of 200

Result page: [1](#) [2](#) [3](#) [4](#) [5](#) [6](#) [7](#) [8](#) [9](#) [10](#) [next](#)

The ACM Portal is published by the Association for Computing Machinery. Copyright © 2005 ACM, Inc.

[Terms of Usage](#) [Privacy Policy](#) [Code of Ethics](#) [Contact Us](#)

Useful downloads:  [Adobe Acrobat](#)  [QuickTime](#)  [Windows Media Player](#)  [Real Player](#)

[Subscribe](#) (Full Service) [Register](#) (Limited Service, Free) [Login](#)Search: ☒ The ACM Digital Library ☐ The Guide

THE ACM DIGITAL LIBRARY

[Feedback](#) [Report a problem](#) [Satisfaction survey](#)

Technique for automatically correcting words in text

Full text [Pdf](#) (6.23 MB)

Source **ACM Computing Surveys (CSUR)** [archive](#)
Volume 24 , Issue 4 (December 1992) [table of contents](#)
Pages: 377 - 439
Year of Publication: 1992
ISSN:0360-0300

Author [Karen Kukich](#) Bellcore, Morristown, NJ

Publisher ACM Press New York, NY, USA

Additional Information: [abstract](#) [references](#) [citations](#) [index terms](#) [review](#) [collaborative colleagues](#) [peer to peer](#)Tools and Actions: [Discussions](#) [Find similar Articles](#) [Review this Article](#)
[Save this Article to a Binder](#) Display Formats: [BibTex](#) [EndNote](#) [ACM Ref](#)DOI Bookmark: Use this link to bookmark this Article: <http://doi.acm.org/10.1145/146370.146380>
[What is a DOI?](#)

↑ ABSTRACT

Research aimed at correcting words in text has focused on three progressively more difficult problems: (1) nonword error detection; (2) isolated-word error correction; and (3) context-dependent word correction. In response to the first problem, efficient pattern-matching and n-gram analysis techniques have been developed for detecting strings that do not appear in a given word list. In response to the second problem, a variety of general and application-specific spelling correction techniques have been developed. Some of them were based on detailed studies of spelling error patterns. In response to the third problem, a few experiments using natural-language-processing tools or statistical-language models have been carried out. This article surveys documented findings on spelling error patterns, provides descriptions of various nonword detection and isolated-word error correction techniques, reviews the state of the art of context-dependent word correction techniques, and discusses research issues related to all three areas of automatic error correction in text.

↑ REFERENCES

Note: OCR errors may be found in this Reference List extracted from the full text article. ACM has opted to expose the complete List rather than only correct and linked references.

1 ABNEY, S. 1990. Rapid incremental parsing with repair. In Proceedings of the 6th New OED Conference: Electronic Text Research (Waterloo, Ontario, Oct. 1990).

2 [Alfred V. Aho, Algorithms for finding patterns in strings, Handbook of theoretical computer science \(vol. A\): algorithms and complexity, MIT Press, Cambridge, MA, 1991](#)